REMARKS/ARGUMENTS

Favorable reconsideration of this application as presently amended and in light of the following discussion is respectfully requested.

Claims 1-37 are pending in the present application with claims 1, 3, 8-17, 19-22, 24, 27-32 and 34 having been amended by the present amendment.

In the outstanding Office Action, claims 1-3, 25 and 37 were rejected under 35 U.S.C. § 103(a) as unpatentable over Bordsen et al.; and claims 4-24 and 26-37 were rejected under 35 U.S.C. § 103(a) as unpatentable over Bordsen et al. in view of Bohannon et al.

Applicants note an Information Disclosure Statement (IDS) was properly filed on March 30, 2004. Accordingly, it is respectfully requested the Examiner acknowledge the consideration of the reference cited therein by issuing an acknowledged PTO-1449 form.

Claims 1-3, 25 and 37 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Bordsen et al. This rejection is respectfully traversed.

Amended claim 1 is directed to a method of logging and recovery in a transaction-processing system. The logging method generates differential log records by applying a bit-wise exclusive-OR (XOR) operation between a before-update image and an after-update image. The recovery method includes redoing updates of committed transactions by applying the bit-wise XOR operation between the differential log records read from the one or more persistent log storage devices and the copy of the database read from the one or more persistent backup storage devices, and undoing updates of uncommitted transactions by applying the bit-wise XOR operation between the differential log records read from the

one or more persistent log storage devices and the copy of the database read from the one or more persistent backup storage devices. Further, the log records are applied in a sequence independently from the order of log creation. Amended independent claims 24 and 34 include similar features in a varying scope.

In a non-limiting example, Fig. 6C illustrates a sequence of log records including log sequences D1, D2 and D3. As shown, the log sequence for D1 is "0111," the log sequence for D2 is "0001" and the log sequence for D3 is "1000." The log records illustrate the differential log records.

For example, Fig. 6A illustrates three transactions T1, T2 and T3. As shown, the initial value is 0, and after the first transaction T1 the value is 7 (a binary value of "0111" as shown in Fig. 6C). After the second transaction T2, the value is 6. The differential value between the value 6 and the value 7 is 1 (a binary value of "0001" as shown for the differential log record D2 in Fig. 6C). After the third transaction T3, the value is 14, which is a difference of 8 between the previous value of 6 (i.e., a binary value of "1000" as shown in Fig. 6C).

As shown in the right side of Fig. 6E, no matter what way the sequence of differential log records are applied in the order of log creation, the proper end value of 14 (after the transaction T3) is determined. For example, consider the first example in which the order of the differential log records used is D1, D3 and D2. Note this order is independent of the order of log creation, which is D1, D2, D3. Thus, to find the last stored value, the

differential log record D1 is XOR'd with D3 and then this value is XOR'd with D2. The below mathematical example illustrates this process.

$(D1 \otimes D3) \otimes D2$

a) $D1 \otimes D3 =$ 0111 (decimal value of 7 for D1)

$\otimes$ 1000 (decimal value of 8 for D3)
1111 (result is decimal value of 15)

b) now XOR the result of "1111" with D2

$1111 \otimes 0001 =$ 1111

$\otimes$ 0001
1110 = decimal value of 14.

Thus, as shown above, the final value of 14 is determined independent of the order of log creation.

However, as shown in the conventional art on the left side of Fig. 6E, the proper value of 14 is not always obtained (i.e., the conventional method has to apply to the log records in a sequence which is the same as the order of log creation).

Because the claimed invention includes a recovery method in which the bit-wise XOR operation between the differential log records and the copy of the database are used, the log records may be applied in a sequence independently from the order of log creation as shown on the right side of Fig. 6E, for example.

The Office Action recognizes Bordsen et al. does not explicitly teach recovering from a failure by replaying the different record in an arbitrary order, which is independent of the order of generation of log records by using bit-wise operations (see item 3 in the Office

Action at page 5), but cites this feature would have been obvious in light of col. 8, lines 35-45 of Bordsen et al., which discusses a Roll-Back recovery and a Roll-Forward recovery method.

However, Applicants note the Roll-Back and Roll-Forward methods in Bordsen et al. are not performed independent of the creation of log records, and do not use differential log records. In more detail, Applicants note the Roll-Forward recovery method in Bordsen et al. is to "redo" the writes of "committed transactions" using log records when the system crashes before the information is wrote to the database. As noted in col. 8, lines 39-42, "[i]n case of a failure during the update of the database, the audit copy can be used to update the database. This recovery action is called a Roll-Forward of the transaction." The audit copy is stored in an audit media.

Further, the Rolled-Back recovery method in Bordsen et al. is to "undo" the writes of uncommitted transactions when the system crashes before a transaction is completed. As noted in col. 8, lines 42-46, "[i]f there is a failure during the write to the audit media, the database is unaltered and there the old before-look image is still good. In this case, the transaction is called a Rolled-Back word recovery, and the update must be retried."

Bordsen et al. also teaches in col. 8, lines 50-56 that "[i]t is then possible to roll a transaction forward or roll a transaction backward during a recovery, as appropriate, because there are <u>two copies</u> in the non-volatile memory, the initial, or before-look copy and the updated (after-look) copy. If there is a failure during the process, recovery can be made because we still have the initial copy" (emphasis added).

That is, the Rolled-Back recovery in Bordsen et al. is accomplished by copying the before-look images into the database rather than doing XOR operations using differential images (as the claimed invention). In Bordsen et al., to enable the undo operation, the before-look images must be kept persistent until the transaction is committed. Thus, Bordsen et al. requires a non-volatile memory for storing the before images as described in col. 5, lines 48-50, which states that "[t]he unmodified before-look image of the locked object is transferred into a non-volatile cash memory."

Further, the Roll-Forward recovery and Roll-Back recovery methods in Bordsen et al. are known to be performed in the same order with the log record creation as referenced by the following references. "The undo/redo recovery policy is: redo all the committed transactions in the order earliest-first, and undo all the incomplete transactions in the order latest-first." (Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, Database Systems: The Complete Book, Pretince Hall, 2002, p. 904).

"It is important in step 1 (undo phase) to process the log backward, to ensure that the resulting state of the database is correct. ... It is important, in this case (redo phase), to process the log forward." (Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, Database Systems Concepts, 4th edition, McGraw Hill, 2002, pp. 659-660).

"In the redo phase, the system replaces updates of all transactions by scanning the log-forward from the last checkpoint. ... In the undo phase, the system rolls-back all transactions in the undo-list. It performs roll-back by scanning the log backward from the end. ... The redo phase ... The actions are repeated in the same order in which the were

carried out; hence, this process is called repeating history." (Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, Database Systems Concepts, 4th edition, McGraw Hill, 2002, pp. 666 and 667).

"During the redo pass, ARIES repeats history, with respect to those updates logged on stable storage ... The next log pass is the undo pass during which all loser transactions' updates are rolled back, in reverse chronological order, in a single sweep of the log." (C. Mohan, Don Haderle, Bruce Lindsay, Hamid Pirahesh, and Peter Schwartz. "ARIES: A Transaction Recovery Method Supporting Fine-Granularity Locking and Partial Rollbacks Using Write-Ahead Logging." ACM Transactions on Database Systems. Volume 17, Number 1, 1992, p. 111).

Bordsen et al. does not teach or suggest that the roll-forward recovery or the roll-backward recovery is done in a different way than the above-conventional method.

In contrast, according to the present invention, the differential logging scheme enables an accurate reconstruction regardless of the order of applying the log records (see the specification at page 6, lines 24-25), even if the order of applying the log records is different from the sequential order of log record creation (see the specification at page 6, line 9). For example, a later-generated log record may be redone ahead of an earlier-generated log record according to the present invention, as shown in the numerical example above.

Thus, it is respectfully submitted that Bordsen et al. does not teach or suggest (1) undoing updates of uncommitted transactions by XOR operations, and (2) applying (redoing or undoing) log records in the order independent of log creation.

Accordingly, it is respectfully submitted independent claims 1, 24 and 34 and each of the claims depending therefrom patentably define over Bordsen et al.

Further, it is respectfully submitted the rejection of claims 4-24 and 26-37 under 35 U.S.C. § 103(a) as unpatentable over Bordsen et al. in view of Bohannon et al. has also been overcome as Bohannon et al. also do not teach or suggest the features recited in the amended independent claims.

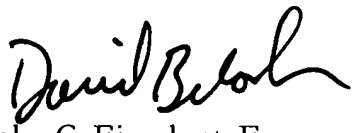
The specification has also been amended to correct minor informalities. It is believed no new matter has been added.

CONCLUSION

In view of the foregoing amendments and remarks, it is respectfully submitted that the application is in condition for allowance. Favorable consideration and prompt allowance are earnestly solicited. If the Examiner believes that any additional changes would place the application in better condition for allowance, the Examiner is invited to contact the undersigned attorney, **David A. Bilodeau**, at the telephone number listed below.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this, concurrent and future replies, including extension of time fees, to Deposit Account 16-0607 and please credit any excess fees to such deposit account.

Respectfully submitted,
FLESHNER & KIM, LLP



John C. Eisenhart, Esq.
Registration No. 37,128
David A. Bilodeau, Esq.
Registration No. 42,325

P.O. Box 221200
Chantilly, VA 20153-1200
703 776-3701 DYK/JCE/DAB:knv
Date: June 17, 2004

Please direct all correspondence to Customer Number 34610